Bioconductor's Computational Ecosystem for Genomic Data Science in Cancer

Multiple

December 24, 2023

Abstract

The Bioconductor project enters its third decade with over two thousand packages for genomic data science, over 100,000 annotation and experiment resources, and a global system for convenient distribution to researchers. The impact of the project on genome biology is attested to by over 60,000 PubMed Central citations and terabytes of content shipped per month. This report provides an overview of cancer genomics resources in Bioconductor. Approaches to cancer data reuse and integration, copy number variation analysis, and methodology for single-cell and spatial transcriptomics are reviewed. Bioconductor's strategies for supporting methods developers and researchers in cancer genomics are evolving along with experimental and computational technologies. All the tools described in this report are backed by regularly maintained learning resources that can be used locally or in cloud computing environments.

Contents

1 Introduction

Computation is a central component of cancer genomics research. Tumor sequencing is the basis of computational investigation of mutational, epigenetic and immunologic processes associated with cancer initiation and progression. Numerous computational workflows have been produced to profile tumor cell transcriptomes and proteomes. New technologies promise to unite sequence-based characterizations with digital histopathology, ultimately driving efforts in molecule design and evaluation to produce patient-centered treatments.

Bioconductor is an open source software project with a 20 year history of uniting biostatisticians, bioinformaticians, and genome researchers in the creation of an ecosystem of data, annotation, and analysis resources for research in genome-scale biology. This paper will review current approaches of the project to advancing cancer genomics. After a brief discussion of basic principles of the Bioconductor project, we will present a "top down" survey of resources useful for cancer bioinformatics. Primary sections address

- how to explore institutionally curated cancer genomics data
- genomic annotation resources relevant to cancer genomics
- analytical workflows
- components for introducing new data or analyses
- pedagogics and workforce development.

The appendix (section 8) of this paper includes descriptions of 69 Bioconductor software packages that use the term "cancer" in their package metadata.

2 Bioconductor principles

2.1 R packages and vignettes

Software tools and data resources in Bioconductor are organized into "R packages". These are collections of folders with data, code (principally R functions), and documentation following a protocol specified in Writing R Extensions. R packages have a DESCRIPTION file with metadata about package contents and provenance. Package structure can be checked for validity using the `R CMD check` facility. Documentation of code and data can be programmatically checked for existence and validity. The DESCRIPTION file for a package specifies its version and also gives precise definition of how an R package may depend upon versions of other packages.

At its inception, Bioconductor introduced a new approach to holistic package documentation called "vignette". Vignettes narrate package operations and include executable code. While R function manual pages describe the operation of individual functions, vignettes illustrate the interoperation of package components.

2.2 R package repositories; repository evolution

Bioconductor software forms a coherent ecosystem that can be checked for consistency of versions of all packages available in a given installation of R. Bioconductor packages may specify dependency on other Bioconductor packages, or packages that are available in the CRAN repository. Bioconductor does not include packages with dependencies on "github-only" packages. Later in this paper we will provide details on package quality assurance that provide a rationale for this restriction.

Major updates to the R language occur annually, and updates are preceded by careful assessment of effects of language change on package operations. These effects can be identified through changes in the output of R CMD check. The Bioconductor ecosystem is updated twice a year, once to coincide with update to R, and once about six months later. The semianual updates reflect the need to track developments in the fast-moving field of genomic data science.

2.3 Package quality assessment; installation consistency

The BiocCheck function is used to provide more stringent assessment of package compliance with basic principles of the Bioconductor ecosystem.

The BiocManager package includes code for checking the consistency and currency of the current collection of installed packages, and for installing or updating packages. This is important in the context of a language and package ecosystem that changes every six months, while analyses may take years to complete. Tools for recreating past package collections are available to assist in reproducing outputs of prior analyses.

2.4 Unifying assay and sample data: SummarizedExperiment and MultiAssayExperiment

Most of the data from genome-scale experiments to be discussed in this chapter are organized in special data containers rooted in the concepts of the SummarizedExperiment class. Briefly, assay data are thought of as occupying a $G \times N$ array, and sample level data occupy an $N \times K$ table. The array and the table are linked together in the SummarizedExperiment; see Figure 1.

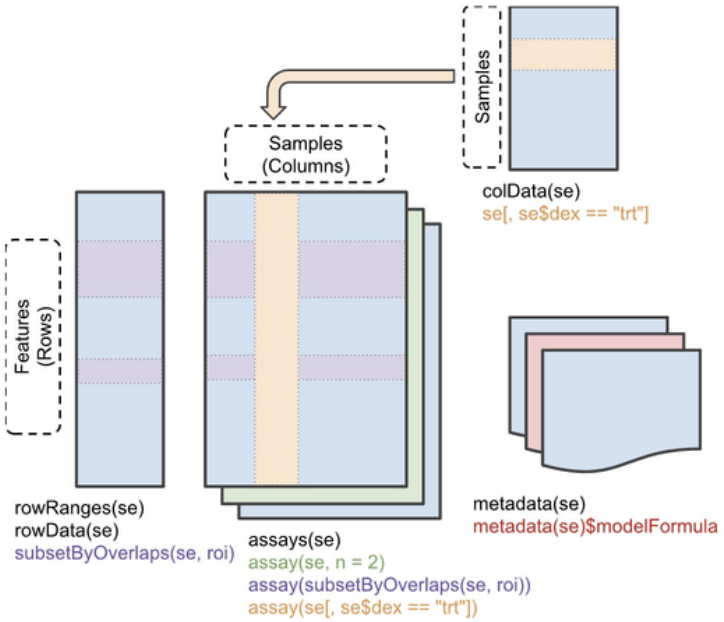

Figure 1: SummarizedExperiment schematic.

Multiple representations of assay results may be managed in this structure, but all assay arrays must have dimensions $G \times N$.

For experiment collections in which the same samples are subjected to multiple genome-scale assays, MultiAssayExperiment containers are used.

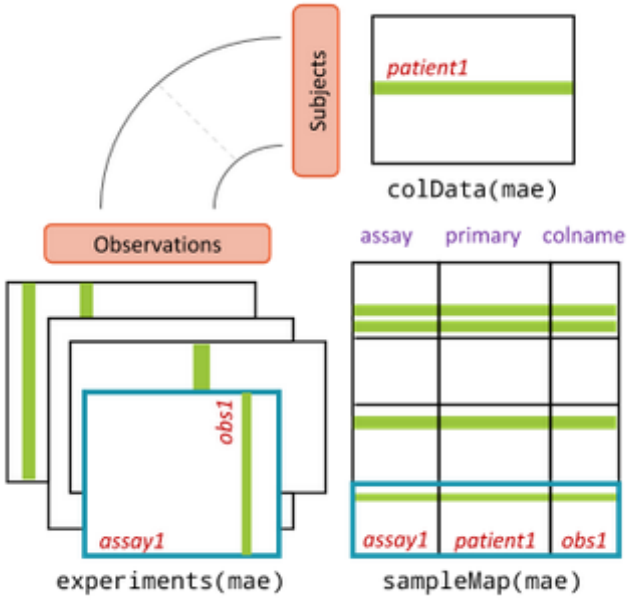

Figure 2: MultiAssayExperiment schematic.

Further details on these data structures will be provided in section 6.

3 Exploring institutionally curated cancer genomics data

3.1 The Cancer Genome Atlas

An overview of Bioconductor's resource for the Cancer Genome Atlas (TCGA) is easy to obtain, with the curatedTCGAData package.

```
library(curatedTCGAData)
tcgatab = curatedTCGAData(version="2.1.1")
```

The first 10 records are in Table 1.

Various conventions are in play in this table. The "title" field is of primary concern. The title string can be decomposed into substrings with interpretation `[tumorcode]_[assay]-[date]_[optional codes]`. The column `ah_id` will be explained in section 4, and column `rdataclass` will be discussed in section 6 below.

3.1.1 Tumor code resolution

There are 33 different tumor types available in TCGA. The decoding of tumor codes for the first ten in alphabetical order is provided in Table 2.

Table 1: First ten records returned by curatedTCGAData::curatedTCGAData().

ah_id	title	file_size	rdataclass
EH4737	ACC_CNASNP-20160128	0.8 Mb	RaggedExperiment
EH4738	ACC_CNVSNP-20160128	0.2 Mb	RaggedExperiment
EH4740	ACC_GISTIC_AllByGene-20160128	0.2 Mb	SummarizedExperiment
EH4741	ACC_GISTIC_Peaks-20160128	0 Mb	RangedSummarizedExperiment
EH4742	ACC_GISTIC_ThresholdedByGene-20160128	0.2 Mb	SummarizedExperiment
EH4744	ACC_Methylation-20160128_assays	239.2 Mb	SummarizedExperiment
EH4745	ACC_Methylation-20160128_se	6 Mb	RaggedExperiment
EH4747	ACC_Mutation-20160128	0.7 Mb	SummarizedExperiment
EH4748	ACC_RNASeq2Gene-20160128	2.7 Mb	SummarizedExperiment
EH4750	ACC_RPPAArray-20160128	0.1 Mb	SummarizedExperiment

Table 2: Decoding TCGA tumor code abbreviations.

Code	Type
ACC	Adrenocortical Carcinoma
BLCA	Bladder Urothelial Carcinoma
BRCA	Breast Invasive Carcinoma
CESC	Cervical Squamous Cell Carcinoma And Endocervical Adenocarcinoma
CHOL	Cholangiocarcinoma
COAD	Colon Adenocarcinoma
DLBC	Lymphoid Neoplasm Diffuse Large B-cell Lymphoma
ESCA	Esophageal Carcinoma
GBM	Glioblastoma Multiforme
HNSC	Head And Neck Squamous Cell Carcinoma

3.1.2 Assay codes and counts

Assays performed on tumors vary across tumor types. For assay types shared between breast cancer, glioblastoma, and lung adenocarcinoma (code LUAD), the numbers of tumor and normal samples available in curatedTCGAData are provided in Table 3.

3.1.3 An example dataset for RNA-seq from glioblastoma multiforme

We obtain normalized RNA-seq data on primary tumor samples for GBM with

```
gbrna = TCGAprimaryTumors(curatedTCGAData("GBM",
    "RNASeq2GeneNorm", dry.run=FALSE, version="2.1.1"))
gbrna
## A MultiAssayExperiment object of 1 listed
##  experiment with a user-defined name and respective class.
##  Containing an ExperimentList class object of length 1:
##  [1] GBM_RNASeq2GeneNorm-20160128: SummarizedExperiment with 18199 rows and 153 columns
## Functionality:
##  experiments() - obtain the ExperimentList instance
##  colData() - the primary/phenotype DataFrame
```

Table 3: Numbers of assays available in TCGA on tumor and normal samples, for breast cancer, glioblastoma, and lung adenocarcinoma.

	BRCA	BRCAnormal	GBM	GBMnormal	LUAD	LUADnormal
CNASNP	1089	1120	577	527	516	579
CNVSNP	1080	1119	577	527	516	579
GISTIC_AllByGene	1080	0	577	0	516	0
GISTIC_Peaks	1080	0	577	0	516	0
GISTIC_ThresholdedByGene	1080	0	577	0	516	0
Mutation	988	5	283	7	230	0
RNASeq2Gene	1093	119	153	13	515	61
RPPAArray	887	50	233	11	365	0
RNASeq2GeneNorm	1093	119	153	13	515	61
Methylation_methyl27	314	29	285	0	65	24
Methylation_methyl450	783	102	140	14	458	34

```
##  sampleMap() - the sample coordination DataFrame
##  `$`, `[`, `[[` - extract colData columns, subset, or experiment
##  *Format() - convert into a long or wide DataFrame
##  assays() - convert ExperimentList to a SimpleList of matrices
##  exportClass() - save data to flat files
```

R functions defined in Bioconductor packages can operate on the variable `gb` to retrieve information of interest. Details on the underlying data structure are given in section 6 below. For most assay types, we think of the quantitative assay information as tabular in nature, with table rows corresponding to genomic features such as genes, and table columns corresponding to samples.

Information on GBM samples employs the `colData` function.

```
dim(colData(gbrna))
## [1]  153 4380
```

For sample level information obtained `colData`, we think of rows as samples, and columns as sample attributes.

3.1.4 Clinical and phenotypic data

TCGA datasets are generally provided as combinations of results for tumor tissue and normal tissue. The determination of a record's sample type is encoded in the sample "barcode". Decoding of sample barcodes is described at the Genomic Data Commons Encyclopedia with specific interpretation of sample types listed separately. The TCGAutils package provides utilities for extracting data on primary tumor samples, excluding samples that may have been taken on normal tissue or metastases.

Clinical and phenotypic data on all TCGA samples are voluminous. For example, there are 2684 fields of sample level data for BRCA samples, and 4380 fields for GBM samples. Many of these fields are meaningfully populated for only a very small minority of samples. To see this for GBM:

```
mean(sapply(colData(gb), function(x) mean(is.na(x))>.90))
## [1] 0.8038813
```

Nevertheless, with careful inspection of fields and contents, clinical data can be extracted and combined with molecular and genetic assay data with modest effort.

The following code chunk illustrates a very crude approach to comparing survival profiles for BRCA, GBM, and LUAD donors.

```
library(survival)
## 0/0 packages newly attached/loaded, see sessionInfo() for details.
getSurv = function(mae) {
 days_on = with(colData(mae), ifelse(is.na(days_to_last_followup),
    days_to_death, days_to_last_followup))
 Surv(days_on, colData(mae)$vital_status)
}
ss = lapply(list(br, gb, lu), getSurv)
codes = c("BRCA", "GBM", "LUAD")
type = factor(rep(codes, sapply(ss,length)))
allsurv = do.call(c, ss)
library(GGally)
## 0/0 packages newly attached/loaded, see sessionInfo() for details.
ggsurv(survfit(allsurv~type))
```

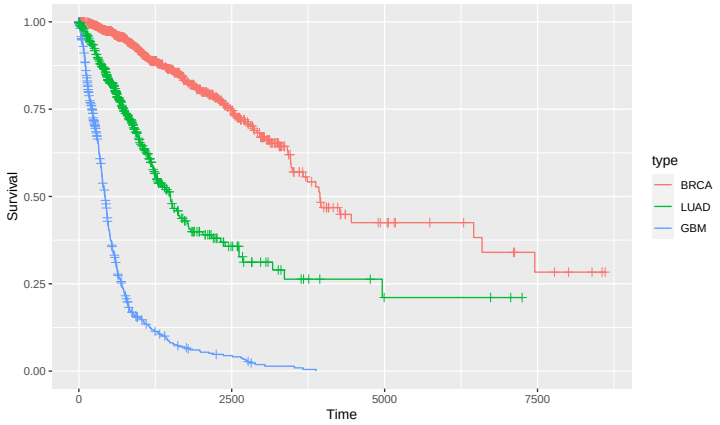

Figure 3: Survival profile extraction from three MultiAssayExperiments produced with curatedTCGAData calls.

3.2 cBioPortal

The cBioPortal user guide defines the goal of the portal to be reducing "the barriers between complex genomic data and cancer researchers by providing rapid, intuitive, and high-quality access to molecular profiles and clinical attributes from large-scale cancer genomics projects, and therefore to empower researchers to translate these rich data sets into biologic insights and clinical applications."

Bioconductor's cBioPortalData package simplifies access to over 300 genomic studies of diverse cancers in cBioPortal. The main unit of data access is the publication. The `cBioPortal` function mediates a connection between an R session and the cBioPortal API. `getStudies` returns a tibble with metadata on all studies.

```
library(cBioPortalData)
cbio = cBioPortal()
allst = getStudies(cbio)
dim(allst)
## [1] 396  13
```

A pruned selection of records from the cBioPortal studies table is given in Table 4.

Table 4: Excerpts from four fields on selected records in the cBioPortal getStudies output.

name	description	studyId	pmid
Ampullary Carcinoma	Exome sequencing ana	ampca_bcm_2016	2680
Hypodiploid Acute Lymphoid Leukemia	Whole genome or exom	all_stjude_2013	2333
Adenoid Cystic Carcinoma of the Breast	Whole exome sequenci	acbc_mskcc_2015	2609
Adenoid Cystic Carcinoma	Whole-exome or whole	acyc_mskcc_2013	2368
Adenoid Cystic Carcinoma	Targeted Sequencing	acyc_fmi_2014	2441
Adenoid Cystic Carcinoma	WGS of 21 salivary A	acyc_mda_2015	2663
Adenoid Cystic Carcinoma	Whole exome sequenci	acyc_sanger_2013	2377
Acute Lymphoblastic Leukemia	Whole-genome and/or	all_stjude_2016	2777
The Angiosarcoma Project - Count Me In	The Angiosarcoma Pro	angs_project_painter_2018	3204
Pediatric Acute Lymphoid Leukemia - Phase II	Whole genome or whol	all_phase2_target_2018_pub	NA

To explore copy number alteration data from a study on angiosarcoma, we find the associated studyId field in `allst` and use the `cBioDataPack` function to retrieve a MultiAssayExperiment:

```
ann = "angs_project_painter_2018"
ang = cBioDataPack(ann)
## Warning in .find_with_xfix(df_colnames, get(paste0(fix, 1)), get(paste0(fix, :
## Multiple prefixes found, using keyword 'region' or taking first one

## Warning in .find_with_xfix(df_colnames, get(paste0(fix, 1)), get(paste0(fix, :
## Multiple prefixes found, using keyword 'region' or taking first one
ang
## A MultiAssayExperiment object of 3 listed
##  experiments with user-defined names and respective classes.
##  Containing an ExperimentList class object of length 3:
##  [1] cna_hg19.seg: RaggedExperiment with 27835 rows and 48 columns
##  [2] cna: SummarizedExperiment with 23109 rows and 48 columns
##  [3] mutations: RaggedExperiment with 24058 rows and 48 columns
## Functionality:
##  experiments() - obtain the ExperimentList instance
##  colData() - the primary/phenotype DataFrame
##  sampleMap() - the sample coordination DataFrame
##  `$`, `[`, `[[` - extract colData columns, subset, or experiment
##  *Format() - convert into a long or wide DataFrame
```

```
##  assays() - convert ExperimentList to a SimpleList of matrices
##  exportClass() - save data to flat files
```

The copy number alteration outcomes are in the `assay` component of the experiment.

```
seg = experiments(ang)[[1]]
colnames(seg) = sapply(strsplit(colnames(seg), "-"), "[", 5)
assay(seg)[1:4,1:4]
##                   DAE1F DACME DADBW DAD34
## 1:12227-955755       71    NA    NA    NA
## 1:957844-1139868     62    NA    NA    NA
## 1:1140874-1471177   167    NA    NA    NA
## 1:1475170-1855370   113    NA    NA    NA
```

The rownames component of this matrix can be transformed to a GenomicRanges instance for concise manipulation.

```
library(GenomicRanges)
## 0/0 packages newly attached/loaded, see sessionInfo() for details.
library(ggplot2)
## 0/0 packages newly attached/loaded, see sessionInfo() for details.
allalt = GRanges(rownames(assay(seg)))
allalt
## GRanges object with 27835 ranges and 0 metadata columns:
##           seqnames            ranges strand
##              <Rle>         <IRanges>  <Rle>
##       [1]        1      12227-955755      *
##       [2]        1    957844-1139868      *
##       [3]        1   1140874-1471177      *
##       [4]        1   1475170-1855370      *
##       [5]        1  1857786-17257894      *
##       ...      ...               ...    ...
##   [27831]       20     68410-1559342      *
##   [27832]       20   1585705-1592359      *
##   [27833]       20  1616247-62904955      *
##   [27834]       21  9907492-48084286      *
##   [27835]       22 16157938-51237572      *
##   -------
##   seqinfo: 22 sequences from an unspecified genome; no seqlengths
```

We'll focus on chromosome 17, where TP53 is found. Regions of genomic alteration are summarized to their midpoints.

```
g17 = allalt[seqnames(allalt)=="17"]
df17 = as(g17, "data.frame")        # for ggplot2
df17$mid = .5*(df17$start+df17$end) # midpoint only
ggplot(df17, aes(x=mid)) + geom_density(bw=.2) + xlab("chr 17 bp")
```

This display shows a strong peak in the vicinity of 7.5 Mb on chromosome 17, near TP53. The display lacks information on the direction of copy number alteration, and on annotation of the genome. These issues will be addressed in later sections.

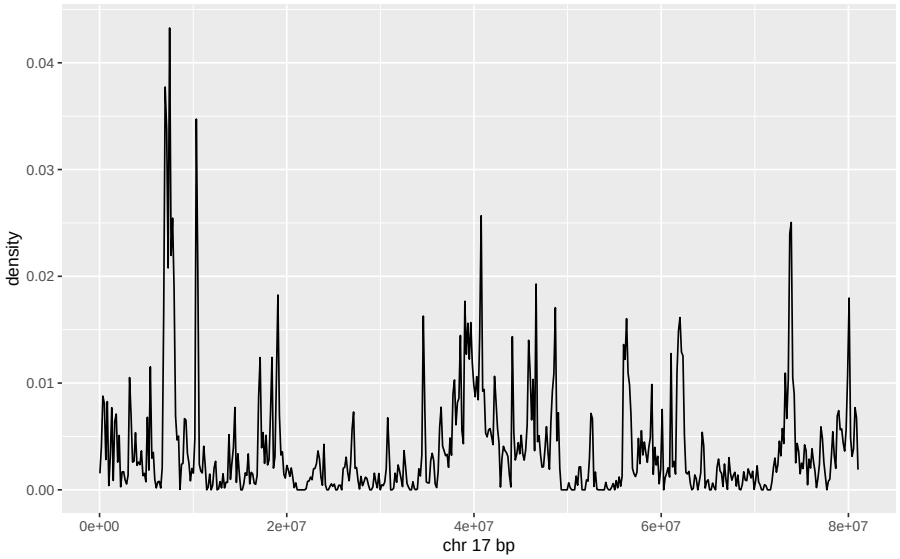

Figure 4: Density of recurrent genomic alterations on chromosome 17 for 48 angiosarcoma patients.

3.3 Resources from NCBI and EMBL

4 Genomic annotation resources relevant to cancer

5 Analytical workflows

6 Components for introducing new data or analyses

6.1 Data structures

Inheritance is a key feature of object-oriented programming (OOP) that allows us to define a new class out of existing classes and add new features, which provides reusability of code. Inheritance carries over attributes and methods defined for base classes; 'Attributes' are variables that are bound in a class. They are used to define behavior and methods for objects of that class. 'Methods' are functions defined within a class that receive an instance of the class, conventionally called self, as the first argument. The attributes defined for a base class will automatically be present in the derived class, and the methods for the base class will work for the derived class. The R programming language has three different class systems: S3, S4, and Reference. Inheritance in S3 classes does not have any fixed definition, and hence attributes of S3 objects can be arbitrary. Derived classes, however, inherit the methods defined for the base class. Inheritance in S4 classes is more structured, and derived classes inherit both attributes and methods of the parent class. Reference classes are similar to S4 classes, but they are mutable and have reference semantics.

S4 classes are used extensively in Bioconductor to create data structures that store complex information, such as biological assay data and metadata, in one or more slots. The entire structure can then be assigned to an R object, and the types of information in each slot of the object are tightly controlled. S4 generics and methods define functions that can be applied to these objects, providing a rich software development infrastructure while ensuring interoperability, reusability, and efficiency.

Bioconductor have established Bioconductor classes to represent different types of biological data. Data and tools distributed through Bioconductor adopt Bioconductor classes, providing convenient methods and improving usability and interoperability within the Bioconductor ecosystem.

Data Types	Bioconductor Classes
Genomic coordinates (1-based, closed interval)	GRanges
Groups of genomic coordinates	GRangesList
Ragged genomic coordinates	RaggedExperiment
Gene sets	GeneSet
Rectangular Features x samples	SummarizedExperiment
Multi-omics data	MultiAssayExperiment
Single-cell data	SingleCellExperiment
Mass spectrometry data	Spectra

The GRanges class represents a collection of genomic ranges and associated annotations. Each element in the vector represents a set genomic ranges in terms of the sequence name (seqnames, typically the chromosome), start and end coordinates (ranges, as an IRanges object), strand (strand, either positive, negative, or unstranded), and optional metadata columns (e.g., exon_id and exon_name in the below).

```
GRanges object with 4 ranges and 2 metadata columns:
      seqnames            ranges strand |   exon_id       exon_name
         <Rle>         <IRanges>  <Rle> | <integer>     <character>
  [1]        X 99883667-99884983      - |    667145 ENSE00001459322
  [2]        X 99885756-99885863      - |    667146 ENSE00000868868
  [3]        X 99887482-99887565      - |    667147 ENSE00000401072
  [4]        X 99887538-99887565      - |    667148 ENSE00001849132
  -------
  seqinfo: 722 sequences (1 circular) from an unspecified genome
```

The GRangesList object serves as a container for genomic features consisting of multiplie ranges that are grouped by a parent features, such as spliced transcripts that are comprised of exons. A GRangesList object behaves like a list and many of the same methods for GRanges objects are available for GRangesList object as well.

The SummarizedExperiment class is a matrix-like container, where rows represent features of interest (e.g., genes, transcripts, exons, etc.) and columns represent samples. The attributes of this object include experimental results (in assays), information on observations (in rowData) and samples (in colData), and additional metadata (in metadata). SummarizedExperiment objects can simultaneouly manage several experimental results as long as they are of the same dimensions. The best benefit of using SummarizedExperiment class is the coordination of the metadata and assays when subsetting. SummarizedExperiment is similar to the historical ExpressionSet class, but more flexible in its row information, allowing both GRanges and DataFrames. ExpressionSet object can be easily converted to SummarizedExperiment.

RangedSummarizedExperiment inherits the SummarizedExperiment class, with the extended capability of storing genomic ranges (as a GRanges or GRangesList object) of interest instead of a DataFrame (S4-class objectcs similar to data.frame) of features in rows.

The MultiAssayExperiment class is modeled after the SummarizedExperiment class.

The SingleCellExperiment classes inherit from the RangedSummarizedExperiment class.

7 Pedagogics and workforce development

Figure 5: Workshop.bioconductor.org schematic.

8 Appendix - Bioconductor software packages with 'cancer' in package description

Package	Description
AMARETTO	Integrating an increasing number of available multi-omics cancer data remains one of the main challenges to improve our understanding of cancer. One of the main challenges is using multi-omics data for identifying novel cancer driver genes. We have developed an algorithm, called AMARETTO, that integrates copy number, DNA methylation and gene expression data to identify a set of driver genes by analyzing cancer samples and connects them to clusters.of co-expressed genes, which we define as modules. We applied AMARETTO in a pancancer setting to identify cancer driver genes and their modules on multiple cancer sites. AMARETTO captures modules enriched in angiogenesis, cell cycle and EMT, and modules that accurately predict survival and molecular subtypes. This allows AMARETTO to identify novel cancer driver genes directing canonical cancer pathways.
BaalChIP	The package offers functions to process multiple ChIP-seq BAM files and detect allele-specific events. Computes allele counts at individual variants (SNPs/SNVs), implements extensive QC steps to remove problematic variants, and utilizes a bayesian framework to identify statistically significant allele- specific events. BaalChIP is able to account for copy number differences between the two alleles, a known phenotypical feature of cancer samples.
bioCancer	This package is a Shiny App to visualize and analyse interactively Multi-Assays of Cancer Genomic Data.
BiocOncoTK	Provide a central interface to various tools for genome-scale analysis of cancer studies.
biodbNci	The biodbNci library is an extension of the biodb framework package. It provides access to biodbNci, a library for connecting to the National Cancer Institute (USA) CACTUS Database. It allows to retrieve entries by their accession number, and run specific web services.

canceR	The package is user friendly interface based on the cgdsr and other modeling packages to explore, compare, and analyse all available Cancer Data (Clinical data, Gene Mutation, Gene Methylation, Gene Expression, Protein Phosphorylation, Copy Number Alteration) hosted by the Computational Biology Center at Memorial-Sloan-Kettering Cancer Center (MSKCC).
cbaf	This package contains functions that allow analysing and comparing omic data across various cancers/cancer subgroups easily. So far, it is compatible with RNA-seq, microRNA-seq, microarray and methylation datasets that are stored on cbioportal.org.
cBioPortalData	The cBioPortalData R package accesses study datasets from the cBio Cancer Genomics Portal. It accesses the data either from the pre-packaged zip / tar files or from the API interface that was recently implemented by the cBioPortal Data Team. The package can provide data in either tabular format or with MultiAssayExperiment object that uses familiar Bioconductor data representations.
cbpManager	This R package provides an R Shiny application that enables the user to generate, manage, and edit data and metadata files suitable for the import in cBioPortal for Cancer Genomics. Create cancer studies and edit its metadata. Upload mutation data of a patient that will be concatenated to the data_mutation_extended.txt file of the study. Create and edit clinical patient data, sample data, and timeline data. Create custom timeline tracks for patients.
ccfindR	A collection of tools for cancer genomic data clustering analyses, including those for single cell RNA-seq. Cell clustering and feature gene selection analysis employ Bayesian (and maximum likelihood) non-negative matrix factorization (NMF) algorithm. Input data set consists of RNA count matrix, gene, and cell bar code annotations. Analysis outputs are factor matrices for multiple ranks and marginal likelihood values for each rank. The package includes utilities for downstream analyses, including meta-gene identification, visualization. and construction of rank-based trees for clusters.
cfDNAPro	cfDNA fragments carry important features for building cancer sample classification ML models, such as fragment size, and fragment end motif etc. Analyzing and visualizing fragment size metrics, as well as other biological features in a curated, standardized, scalable, well-documented, and reproducible way might be time intensive. This package intends to resolve these problems and simplify the process. It offers two sets of functions for cfDNA feature characterization and visualization.
cfTools	The cfTools R package provides methods for cell-free DNA (cfDNA) methylation data analysis to facilitate cfDNA-based studies. Given the methylation sequencing data of a cfDNA sample, for each cancer marker or tissue marker, we deconvolve the tumor-derived or tissue-specific reads from all reads falling in the marker region. Our read-based deconvolution algorithm exploits the pervasiveness of DNA methylation for signal enhancement, therefore can sensitively identify a trace amount of tumor-specific or tissue-specific cfDNA in plasma. cfTools provides functions for (1) cancer detection: sensitively detect tumor-derived cfDNA and estimate the tumor-derived cfDNA fraction (tumor burden); (2) tissue deconvolution: infer the tissue type composition and the cfDNA fraction of multiple tissue types for a plasma cfDNA sample. These functions can serve as foundations for more advanced cfDNA-based studies, including cancer diagnosis and disease monitoring.
CIMICE	CIMICE is a tool in the field of tumor phylogenetics and its goal is to build a Markov Chain (called Cancer Progression Markov Chain, CPMC) in order to model tumor subtypes evolution. The input.of CIMICE is a Mutational Matrix, so a boolean matrix representing altered genes in a collection of samples. These samples are assumed to be obtained with single-cell DNA analysis techniques and the tool is specifically written to use the peculiarities of this data for the CMPC construction.

compSPOT	Clonal cell groups share common mutations within cancer, precancer, and even clinically normal appearing tissues. The frequency and location of these mutations may predict prognosis and cancer risk. It has also been well established that certain genomic regions have increased sensitivity to acquiring mutations. Mutation-sensitive genomic regions may therefore serve as markers for predicting cancer risk. This package contains multiple functions to establish significantly mutated hotspots, compare hotspot mutation burden between samples, and perform exploratory data analysis of the correlation between hotspot mutation burden and personal risk factors for cancer, such as age, gender, and history of carcinogen exposure. This package allows users to identify robust genomic markers to help establish cancer risk.
consensusOV	This package implements four major subtype classifiers for high-grade serous (HGS) ovarian cancer as described by Helland et al. (PLoS One, 2011), Bentink et al. (PLoS One, 2012), Verhaak et al. (J Clin Invest, 2013), and Konecny et al. (J Natl Cancer Inst, 2014). In addition, the package implements a consensus classifier, which consolidates and improves on the robustness of the proposed subtype classifiers, thereby providing reliable stratification of patients with HGS ovarian tumors of clearly defined subtype.
copa	COPA is a method to find genes that undergo recurrent fusion in a given cancer type by finding pairs of genes that have mutually exclusive outlier profiles.
dce	Compute differential causal effects (dce) on (biological) networks. Given observational samples from a control experiment and non-control (e.g., cancer) for two genes A and B, we can compute differential causal effects with a (generalized) linear regression. If the causal effect of gene A on gene B in the control samples is different from the causal effect in the non-control samples the dce will differ from zero. We regularize the dce computation by the inclusion of prior network information from pathway databases such as KEGG.
DepInfeR	DepInfeR integrates two experimentally accessible input data matrices: the drug sensitivity profiles of cancer cell lines or primary tumors ex-vivo (X), and the drug affinities of a set of proteins (Y), to infer a matrix of molecular protein dependencies of the cancers (ß). DepInfeR deconvolutes the protein inhibition effect on the viability phenotype by using regularized multivariate linear regression. It assigns a "dependence coefficient" to each protein and each sample, and therefore could be used to gain a causal and accurate understanding of functional consequences of genomic aberrations in a heterogeneous disease, as well as to guide the choice of pharmacological intervention for a specific cancer type, sub-type, or an individual patient. For more information, please read out preprint on bioRxiv: https://doi.org/10.1101/2022.01.11.475864.
DriverNet	DriverNet is a package to predict functional important driver genes in cancer by integrating genome data (mutation and copy number variation data) and transcriptome data (gene expression data). The different kinds of data are combined by an influence graph, which is a gene-gene interaction network deduced from pathway data. A greedy algorithm is used to find the possible driver genes, which may mutated in a larger number of patients and these mutations will push the gene expression values of the connected genes to some extreme values.
easier	This package provides a workflow for the use of EaSIeR tool, developed to assess patients' likelihood to respond to ICB therapies providing just the patients' RNA-seq data as input. We integrate RNA-seq data with different types of prior knowledge to extract quantitative descriptors of the tumor microenvironment from several points of view, including composition of the immune repertoire, and activity of intra- and extra-cellular communications. Then, we use multi-task machine learning trained in TCGA data to identify how these descriptors can simultaneously predict several state-of-the-art hallmarks of anti-cancer immune response. In this way we derive cancer-specific models and identify cancer-specific systems biomarkers of immune response. These biomarkers have been experimentally validated in the literature and the performance of EaSIeR predictions has been validated using independent datasets form four different cancer types with patients treated with anti-PD1 or anti-PDL1 therapy.

GDCRNATools	This is an easy-to-use package for downloading, organizing, and integrative analyzing RNA expression data in GDC with an emphasis on deciphering the lncRNA-mRNA related ceRNA regulatory network in cancer. Three databases of lncRNA-miRNA interactions including spongeScan, starBase, and miRcode, as well as three databases of mRNA-miRNA interactions including miRTarBase, starBase, and miRcode are incorporated into the package for ceRNAs network construction. limma, edgeR, and DESeq2 can be used to identify differentially expressed genes/miRNAs. Functional enrichment analyses including GO, KEGG, and DO can be performed based on the clusterProfiler and DO packages. Both univariate CoxPH and KM survival analyses of multiple genes can be implemented in the package. Besides some routine visualization functions such as volcano plot, bar plot, and KM plot, a few simply shiny apps are developed to facilitate visualization of results on a local webpage.
genefu	This package contains functions implementing various tasks usually required by gene expression analysis, especially in breast cancer studies: gene mapping between different microarray platforms. identification of molecular subtypes, implementation of published gene signatures. gene selection, and survival analysis.
GeoTcgaData	Gene Expression Omnibus(GEO) and The Cancer Genome Atlas (TCGA) provide us with a wealth of data, such as RNA-seq, DNA Methylation, SNP and Copy number variation data. It's easy to download data from TCGA using the gdc tool, but processing these data.into a format suitable for bioinformatics analysis requires more work. This R package was developed to handle these data.
INDEED	An R package for integrated differential expression and differential network analysis based on omic data for cancer biomarker discovery. Both correlation and partial correlation can be used to generate differential network to aid the traditional differential expression analysis to identify changes between biomolecules on both their expression and pairwise association levels. A detailed description of the methodology has been published in Methods journal (PMID: 27592383). An interactive visualization feature allows for the exploration and selection of candidate biomarkers.
iPath	iPath is the Bioconductor package used for calculating personalized pathway score and test the association with survival outcomes. Abundant single-gene biomarkers have been identified and used in the clinics. However, hundreds of oncogenes or tumor-suppressor genes are involved during the process of tumorigenesis. We believe individual-level expression patterns of pre-defined pathways or gene sets are better biomarkers than single genes. In this study, we devised a computational method named iPath to identify prognostic biomarker pathways, one sample at a time. To test its utility, we conducted a pan-cancer analysis across 14 cancer types from The Cancer Genome Atlas and demonstrated that iPath is capable of identifying highly predictive biomarkers for clinical outcomes, including overall survival, tumor subtypes, and tumor stage classifications. We found that pathway-based biomarkers are more robust and effective than single genes.
LACE	LACE is an algorithmic framework that processes single-cell somatic mutation profiles from cancer samples collected at different time points and in distinct experimental settings, to produce longitudinal models of cancer evolution. The approach solves a Boolean Matrix Factorization problem with phylogenetic constraints, by maximizing a weighed likelihood function computed on multiple time points.
macat	This library contains functions to investigate links between differential gene expression and the chromosomal localization of the genes. MACAT is motivated by the common observation of phenomena involving large chromosomal regions in tumor cells. MACAT is the implementation of a statistical approach for identifying significantly differentially expressed chromosome regions. The functions have been tested on a publicly available data set about acute lymphoblastic leukemia (Yeoh et al.Cancer Cell 2002), which is provided in the library 'stjudem'.
maftools	Analyze and visualize Mutation Annotation Format (MAF) files from large scale sequencing studies. This package provides various functions to perform most commonly used analyses in cancer genomics and to create feature rich customizable visualzations with minimal effort.

mastR	mastR is an R package designed for automated screening of signatures of interest for specific research questions. The package is developed for generating refined lists of signature genes from multiple group comparisons based on the results from edgeR and limma differential expression (DE) analysis workflow. It also takes into account the background noise of tissue-specificity, which is often ignored by other marker generation tools. This package is particularly useful for the identification of group markers in various biological and medical applications, including cancer research and developmental biology.
MethylMix	MethylMix is an algorithm implemented to identify hyper and hypomethylated genes for a disease. MethylMix is based on a beta mixture model to identify methylation states and compares them with the normal DNA methylation state. MethylMix uses a novel statistic, the Differential Methylation value or DM-value defined as the difference of a methylation state with the normal methylation state. Finally, matched gene expression data is used to identify, besides differential, functional methylation states by focusing on methylation changes that effect gene expression. References: Gevaert 0. MethylMix: an R package for identifying DNA methylation-driven genes. Bioinformatics (Oxford, England). 2015;31(11):1839-41. doi:10.1093/bioinformatics/btv020. Gevaert O, Tibshirani R, Plevritis SK. Pancancer analysis of DNA methylation-driven genes using MethylMix. Genome Biology. 2015;16(1):17. doi:10.1186/s13059-014-0579-8.
Moonlight2R	The understanding of cancer mechanism requires the identification of genes playing a role in the development of the pathology and the characterization of their role (notably oncogenes and tumor suppressors). We present an updated version of the R/bioconductor package called MoonlightR, namely Moonlight2R, which returns a list of candidate driver genes for specific cancer types on the basis of omics data integration. The Moonlight framework contains a primary layer where gene expression data and information about biological processes are integrated to predict genes called oncogenic mediators, divided into putative tumor suppressors and putative oncogenes. This is done through functional enrichment analyses, gene regulatory networks and upstream regulator analyses to score the importance of well-known biological processes with respect to the studied cancer type. By evaluating the effect of the oncogenic mediators on biological processes or through random forests, the primary layer predicts two putative roles for the oncogenic mediators: i) tumor suppressor genes (TSGs) and ii) oncogenes (OCGs). As gene expression data alone is not enough to explain the deregulation of the genes, a second layer of evidence is needed. We have automated the integration of a secondary mutational layer through new functionalities in Moonlight2R. These functionalities analyze mutations in the cancer cohort and classifies these into driver and passenger mutations using the driver mutation prediction tool, CScape-somatic. Those oncogenic mediators with at least one driver mutation are retained as the driver genes. As a consequence, this methodology does not only identify genes playing a dual role (e.g. TSG in one cancer type and OCG in another) but also helps in elucidating the biological processes underlying their specific roles. In particular, Moonlight2R can be used to discover OCGs and TSGs in the same cancer type. This may for instance help in answering the question whether some genes change role between early stages (I, II) and late stages (III, IV). In the future, this analysis could be useful to determine the causes of different resistances to chemotherapeutic treatments.

MoonlightR	Motivation: The understanding of cancer mechanism requires the identification of genes playing a role in the development of the pathology and the characterization of their role (notably oncogenes and tumor suppressors). Results: We present an R/bioconductor package called MoonlightR which returns a list of candidate driver genes for specific cancer types on the basis of TCGA expression data. The method first infers gene regulatory networks and then carries out a functional enrichment analysis (FEA) (implementing an upstream regulator analysis, URA) to score the importance of well-known biological processes with respect to the studied cancer type. Eventually, by means of random forests, MoonlightR predicts two specific roles for the candidate driver genes: i) tumor suppressor genes (TSGs) and ii) oncogenes (OCGs). As a consequence, this methodology does not only identify genes playing a dual role (e.g. TSG in one cancer type and OCG in another) but also helps in elucidating the biological processes underlying their specific roles. In particular, MoonlightR can be used to discover OCGs and TSGs in the same cancer type. This may help in answering the question whether some genes change role between early stages (I, II) and late stages (III, IV) in breast cancer. In the future, this analysis could be useful to determine the causes of different resistances to chemotherapeutic treatments.
NoRCE	While some non-coding RNAs (ncRNAs) are assigned critical regulatory roles, most remain functionally uncharacterized. This presents a challenge whenever an interesting set of ncRNAs needs to be analyzed in a functional context. Transcripts located close-by on the genome are often regulated together. This genomic proximity on the sequence can hint to a functional association. We present a tool, NoRCE, that performs cis enrichment analysis for a given set of ncRNAs. Enrichment is carried out using the functional annotations of the coding genes located proximal to the input ncRNAs. Other biologically relevant information such as topologically associating domain (TAD) boundaries, co-expression patterns, and miRNA target prediction information can be incorporated to conduct a richer enrichment analysis. To this end, NoRCE includes several relevant datasets as part of its data repository, including cell-line specific TAD boundaries, functional gene sets, and expression data for coding & ncRNAs specific to cancer. Additionally, the users can utilize custom data files in their investigation. Enrichment results can be retrieved in a tabular format or visualized in several different ways. NoRCE is currently available for the following species: human, mouse, rat, zebrafish, fruit fly, worm, and yeast.
octad	OCTAD provides a platform for virtually screening compounds targeting precise cancer patient groups. The essential idea is to identify drugs that reverse the gene expression signature of disease by tamping down over-expressed genes and stimulating weakly expressed ones. The package offers deep-learning based reference tissue selection, disease gene expression signature creation, pathway enrichment analysis, drug reversal potency scoring, cancer cell line selection, drug enrichment analysis and in silico hit validation. It currently covers ~20,000 patient tissue samples covering 50 cancer types, and expression profiles for ~12,000 distinct compounds.
oncoscanR	The software uses the copy number segments from a text file and identifies all chromosome arms that are globally altered and computes various genome-wide scores. The following HRD scores (characteristic of BRCA-mutated cancers) are included: LST, HR-LOH, nLST and gLOH. the package is tailored for the ThermoFisher Oncoscan assay analyzed with their Chromosome Alteration Suite (ChAS) but can be adapted to any input.
OncoScore	OncoScore is a tool to measure the association of genes to cancer based on citation frequencies in biomedical literature. The score is evaluated from PubMed literature by dynamically updatable web queries.

OncoSimulR	Functions for forward population genetic simulation in asexual populations, with special focus on cancer progression. Fitness can be an arbitrary function of genetic interactions between multiple genes or modules of genes, including epistasis, order restrictions in mutation accumulation, and order effects. Fitness (including just birth, just death, or both birth and death) can also be a function of the relative and absolute frequencies of other genotypes (i.e., frequency-dependent fitness). Mutation rates can differ between genes, and we can include mutator/antimutator genes (to model mutator phenotypes). Simulating multi-species scenarios and therapeutic interventions, including adaptive therapy, is also possible. Simulations use continuous-time models and can include driver and passenger genes and modules. Also included are functions for: simulating random DAGs of the type found in Oncogenetic Trees, Conjunctive Bayesian Networks, and other cancer progression models; plotting and sampling from single or multiple realizations of the simulations, including single-cell sampling; plotting the parent-child relationships of the clones; generating random fitness landscapes (Rough Mount Fuji, House of Cards, additive, NK, Ising, and Eggbox models) and plotting them.
oppar	The R implementation of mCOPA package published by Wang et al. (2012). Oppar provides methods for Cancer Outlier profile Analysis. Although initially developed to detect outlier genes in cancer studies, methods presented in oppar can be used for outlier profile analysis in general. In addition, tools are provided for gene set enrichment and pathway analysis.
ORFhunteR	The ORFhunteR package is a R and C++ library for an automatic determination and annotation of open reading frames (ORF) in a large set of RNA molecules. It efficiently implements the machine learning model based on vectorization of nucleotide sequences and the random forest classification algorithm. The ORFhunteR package consists of a set of functions written in the R language in conjunction with C++. The efficiency of the package was confirmed by the examples of the analysis of RNA molecules from the NCBI RefSeq and Ensembl databases. The package can be used in basic and applied biomedical research related to the study of the transcriptome of normal as well as altered (for example, cancer) human cells.
OutSplice	An easy to use tool that can compare splicing events in tumor and normal tissue samples using either a user generated matrix, or data from The Cancer Genome Atlas (TCGA). This package generates a matrix of splicing outliers that are significantly over or underexpressed in tumors samples compared to normal denoted by chromosome location. The package also will calculate the splicing burden in each tumor and characterize the types of splicing events that occur.
pathifier	Pathifier is an algorithm that infers pathway deregulation scores for each tumor sample on the basis of expression data. This score is determined, in a context-specific manner, for every particular dataset and type of cancer that is being investigated. The algorithm transforms gene-level information into pathway-level information, generating a compact and biologically relevant representation of each sample.
paxtoolsr	The package provides a set of R functions for interacting with BioPAX OWL files using Paxtools and the querying Pathway Commons (PC) molecular interaction database. Pathway Commons is a project by the Memorial Sloan-Kettering Cancer Center (MSKCC), Dana-Farber Cancer Institute (DFCI), and the University of Toronto. Pathway Commons databases include: BIND, BioGRID, CORUM, CTD, DIP, DrugBank, HPRD, HumanCyc, IntAct, KEGG, MirTarBase, Panther, PhosphoSitePlus, Reactome, RECON, TRANSFAC.

PDATK	Pancreatic ductal adenocarcinoma (PDA) has a relatively poor prognosis and is one of the most lethal cancers. Molecular classification of gene expression profiles holds the potential to identify meaningful subtypes which can inform therapeutic strategy in the clinical setting. The Pancreatic Cancer Adenocarcinoma Tool-Kit (PDATK) provides an S4 class-based interface for performing unsupervised subtype discovery, cross-cohort meta-clustering, gene-expression-based classification, and subsequent survival analysis to identify prognostically useful subtypes in pancreatic cancer and beyond. Two novel methods, Consensus Subtypes in Pancreatic Cancer (CSPC) and Pancreatic Cancer Overall Survival Predictor (PCOSP) are included for consensus-based meta-clustering and overall-survival prediction, respectively. Additionally, four published subtype classifiers and three published prognostic gene signatures are included to allow users to easily recreate published results, apply existing classifiers to new data, and benchmark the relative performance of new methods. The use of existing Bioconductor classes as input to all PDATK classes and methods enables integration with existing Bioconductor datasets, including the 21 pancreatic cancer patient cohorts available in the MetaGxPancreas data package. PDATK has been used to replicate results from Sandhu et al (2019) [https://doi.org/10.1200/cci.18.00102] and an additional paper is in the works using CSPC to validate subtypes from the included published classifiers, both of which use the data available in MetaGxPancreas. The inclusion of subtype centroids and prognostic gene signatures from these and other publications will enable researchers and clinicians to classify novel patient gene expression data, allowing the direct clinical application of the classifiers included in PDATK. Overall, PDATK provides a rich set of tools to identify and validate useful prognostic and molecular subtypes based on gene-expression data, benchmark new classifiers against existing ones, and apply discovered classifiers on novel patient data to inform clinical decision making.
PharmacoGx	Contains a set of functions to perform large-scale analysis of pharmaco-genomic data. These include the PharmacoSet object for storing the results of pharmacogenomic experiments, as well as a number of functions for computing common summaries of drug-dose response and correlating them with the molecular features in a cancer cell-line.
psichomics	Interactive R package with an intuitive Shiny-based graphical interface for alternative splicing quantification and integrative analyses of alternative splicing and gene expression based on The Cancer Genome Atlas (TCGA), the Genotype-Tissue Expression project (GTEx), Sequence Read Archive (SRA) and user-provided data. The tool interactively performs survival, dimensionality reduction and median- and variance-based differential splicing and gene expression analyses that benefit from the incorporation of clinical and molecular sample-associated features (such as tumour stage or survival). Interactive visual access to genomic mapping and functional annotation of selected alternative splicing events is also included.
RadioGx	Computational tool box for radio-genomic analysis which integrates radio-response data, radio-biological modelling and comprehensive cell line annotations for hundreds of cancer cell lines. The 'RadioSet' class enables creation and manipulation of standardized datasets including information about cancer cells lines, radio-response assays and dose-response indicators. Included methods allow fitting and plotting dose-response data using established radio-biological models along with quality control to validate results. Additional functions related to fitting and plotting dose response curves, quantifying statistical correlation and calculating area under the curve (AUC) or survival fraction (SF) are included. For more details please see the included documentation, references, as well as: Manem, V. et al (2018) <doi:10.1101/449793>.
RAIDS	This package implements specialized algorithms that enable genetic ancestry inference from various cancer sequences sources (RNA, Exome and Whole-Genome sequences). This package also implements a simulation algorithm that generates synthetic cancer-derived data. This code and analysis pipeline was designed and developed for the following publication: Belleau, P et al. Genetic Ancestry Inference from Cancer-Derived Molecular Data across Genomic and Transcriptomic Platforms. Cancer Res 1 January 2023; 83 (1): 49–58.

rcellminer	The NCI-60 cancer cell line panel has been used over the course of several decades as an anti-cancer drug screen. This panel was developed as part of the Developmental Therapeutics Program (DTP, http://dtp.nci.nih.gov/) of the U.S. National Cancer Institute (NCI). Thousands of compounds have been tested on the NCI-60, which have been extensively characterized by many platforms for gene and protein expression, copy number, mutation, and others (Reinhold, et al., 2012). The purpose of the CellMiner project (http://discover.nci.nih.gov/ cellminer) has been to integrate data from multiple platforms used to analyze the NCI-60 and to provide a powerful suite of tools for exploration of NCI-60 data.
RESOLVE	Cancer is a genetic disease caused by somatic mutations in genes controlling key biological functions such as cellular growth and division. Such mutations may arise both through cell-intrinsic and exogenous processes, generating characteristic mutational patterns over the genome named mutational signatures. The study of mutational signatures have become a standard component of modern genomics studies, since it can reveal which (environmental and endogenous) mutagenic processes are active in a tumor, and may highlight markers for therapeutic response. Mutational signatures computational analysis presents many pitfalls. First, the task of determining the number of signatures is very complex and depends on heuristics. Second, several signatures have no clear etiology, casting doubt on them being computational artifacts rather than due to mutagenic processes. Last, approaches for signatures assignment are greatly influenced by the set of signatures used for the analysis. To overcome these limitations, we developed RESOLVE (Robust EStimation Of mutationaL signatures Via rEgularization), a framework that allows the efficient extraction and assignment of mutational signatures. RESOLVE implements a novel algorithm that enables (i) the efficient extraction, (ii) exposure estimation, and (iii) confidence assessment during the computational inference of mutational signatures.
RLassoCox	RLassoCox is a package that implements the RLasso-Cox model proposed by Wei Liu. The RLasso-Cox model integrates gene interaction information into the Lasso-Cox model for accurate survival prediction and survival biomarker discovery. It is based on the hypothesis that topologically important genes in the gene interaction network tend to have stable expression changes. The RLasso-Cox model uses random walk to evaluate the topological weight of genes, and then highlights topologically important genes to improve the generalization ability of the Lasso-Cox model. The RLasso-Cox model has the advantage of identifying small gene sets with high prognostic performance on independent datasets, which may play an important role in identifying robust survival biomarkers for various cancer types.
RTCGA	The Cancer Genome Atlas (TCGA) Data Portal provides a platform for researchers to search, download, and analyze data sets generated by TCGA. It contains clinical information, genomic characterization data, and high level sequence analysis of the tumor genomes. The key is to understand genomics to improve cancer care. RTCGA package offers download and integration of the variety and volume of TCGA data using patient barcode key, what enables easier data possession. This may have an benefcial infuence on impact on development of science and improvement of patients' treatment. Furthermore, RTCGA package transforms TCGA data to tidy form which is convenient to use.
RTCGAToolbox	Managing data from large scale projects such as The Cancer Genome Atlas (TCGA) for further analysis is an important and time consuming step for research projects. Several efforts, such as Firehose project, make TCGA pre-processed data publicly available via web services and data portals but it requires managing, downloading and preparing the data for following steps. We developed an open source and extensible R based data client for Firehose pre-processed data and demonstrated its use with sample case studies. Results showed that RTCGAToolbox could improve data management for researchers who are interested with TCGA data. In addition, it can be integrated with other analysis pipelines for following data analysis.

SCFA	Subtyping via Consensus Factor Analysis (SCFA) can efficiently remove noisy signals from consistent molecular patterns in multi-omics data. SCFA first uses an autoencoder to select only important features and then repeatedly performs factor analysis to represent the data with different numbers of factors. Using these representations, it can reliably identify cancer subtypes and accurately predict risk scores of patients.
SCOPE	Whole genome single-cell DNA sequencing (scDNA-seq) enables characterization of copy number profiles at the cellular level. This circumvents the averaging effects associated with bulk-tissue sequencing and has increased resolution yet decreased ambiguity in deconvolving cancer subclones and elucidating cancer evolutionary history. ScDNA-seq data is, however, sparse, noisy, and highly variable even within a homogeneous cell population, due to the biases and artifacts that are introduced during the library preparation and sequencing procedure. Here, we propose SCOPE, a normalization and copy number estimation method for scDNA-seq data. The distinguishing features of SCOPE include: (i) utilization of cell-specific Gini coefficients for quality controls and for identification of normal/diploid cells, which are further used as negative control samples in a Poisson latent factor model for normalization; (ii) modeling of GC content bias using an expectation-maximization algorithm embedded in the Poisson generalized linear models, which accounts for the different copy number states along the genome; (iii) a cross-sample iterative segmentation procedure to identify breakpoints that are shared across cells from the same genetic background.
seq.hotSPOT	seq.hotSPOT provides a resource for designing effective sequencing panels to help improve mutation capture efficacy for ultradeep sequencing projects. Using SNV datasets, this package designs custom panels for any tissue of interest and identify the genomic regions likely to contain the most mutations. Establishing efficient targeted sequencing panels can allow researchers to study mutation burden in tissues at high depth without the economic burden of whole-exome or whole-genome sequencing. This tool was developed to make high-depth sequencing panels to study low-frequency clonal mutations in clinically normal and cancerous tissues.
seqCNA	Copy number analysis of high-throughput sequencing cancer data with fast summarization, extensive filtering and improved normalization
sevenbridges	R client and utilities for Seven Bridges platform API, from Cancer Genomics Cloud to other Seven Bridges supported platforms.
SigCheck	While gene signatures are frequently used to predict phenotypes (e.g. predict prognosis of cancer patients), it it not always clear how optimal or meaningful they are (cf David Venet, Jacques E. Dumont, and Vincent Detours' paper "Most Random Gene Expression Signatures Are Significantly Associated with Breast Cancer Outcome"). Based on suggestions in that paper, SigCheck accepts a data set (as an ExpressionSet) and a gene signature, and compares its performance on survival and/or classification tasks against a) random gene signatures of the same length; b) known, related and unrelated gene signatures; and c) permuted data and/or metadata.
signeR	The signeR package provides an empirical Bayesian approach to mutational signature discovery. It is designed to analyze single nucleotide variation (SNV) counts in cancer genomes, but can also be applied to other features as well. Functionalities to characterize signatures or genome samples according to exposure patterns are also provided.
signifinder	signifinder is an R package for computing and exploring a compendium of tumor signatures. It allows to compute a variety of signatures, based on gene expression values, and return single-sample scores. Currently, signifinder contains 46 distinct signatures collected from the literature, relating to multiple tumors and multiple cancer processes.
supersigs	Generate SuperSigs (supervised mutational signatures) from single nucleotide variants in the cancer genome. Functions included in the package allow the user to learn supervised mutational signatures from their data and apply them to new data. The methodology is based on the one described in Afsari (2021, ELife).

TRONCO	The TRONCO (TRanslational ONCOlogy) R package collects algorithms to infer progression models via the approach of Suppes-Bayes Causal Network, both from an ensemble of tumors (cross-sectional samples) and within an individual patient (multi-region or single-cell samples). The package provides parallel implementation of algorithms that process binary matrices where each row represents a tumor sample and each column a single-nucleotide or a structural variant driving the progression; a 0/1 value models the absence/presence of that alteration in the sample. The tool can import data from plain, MAF or GISTIC format files, and can fetch it from the cBioPortal for cancer genomics. Functions for data manipulation and visualization are provided, as well as functions to import/export such data to other bioinformatics tools for, e.g, clustering or detection of mutually exclusive alterations. Inferred models can be visualized and tested for their confidence via bootstrap and cross-validation. TRONCO is used for the implementation of the Pipeline for Cancer Inference (PICNIC).
Uniquorn	'Uniquorn' enables users to identify cancer cell lines. Cancer cell line misidentification and cross-contamination reprents a significant challenge for cancer researchers. The identification is vital and in the frame of this package based on the locations/ loci of somatic and germline mutations/ variations. The input format is vcf/ vcf.gz and the files have to contain a single cancer cell line sample (i.e. a single member/genotype/gt column in the vcf file).
ZygosityPredictor	The ZygosityPredictor allows to predict how many copies of a gene are affected by small variants. In addition to the basic calculations of the affected copy number of a variant, the Zygosity-Predictor can integrate the influence of several variants on a gene and ultimately make a statement if and how many wild-type copies of the gene are left. This information proves to be of particular use in the context of translational medicine. For example, in cancer genomes, the Zygosity-Predictor can address whether unmutated copies of tumor-suppressor genes are present. Beyond this, it is possible to make this statement for all genes of an organism. The Zygosity-Predictor was primarily developed to handle SNVs and INDELs (later addressed as small-variants) of somatic and germline origin. In order not to overlook severe effects outside of the small-variant context, it has been extended with the assessment of large scale deletions, which cause losses of whole genes or parts of them.
CancerInSilico	The CancerInSilico package provides an R interface for running mathematical models of tumor progresson and generating gene expression data from the results. This package has the underlying models implemented in C++ and the output and analysis features implemented in R.
CancerSubtypes	CancerSubtypes integrates the current common computational biology methods for cancer subtypes identification and provides a standardized framework for cancer subtype analysis based multi-omics data, such as gene expression, miRNA expression, DNA methylation and others.
IRISFGM	Single-cell RNA-Seq data is useful in discovering cell heterogeneity and signature genes in specific cell populations in cancer and other complex diseases. Specifically, the investigation of functional gene modules (FGM) can help to understand gene interactive networks and complex biological processes. QUBIC2 is recognized as one of the most efficient and effective tools for FGM identification from scRNA-Seq data. However, its availability is limited to a C implementation, and its applicative power is affected by only a few downstream analyses functionalities. We developed an R package named IRIS-FGM (integrative scRNA-Seq interpretation system for functional gene module analysis) to support the investigation of FGMs and cell clustering using scRNA-Seq data. Empowered by QUBIC2, IRIS-FGM can identify co-expressed and co-regulated FGMs, predict types/clusters, identify differentially expressed genes, and perform functional enrichment analysis. It is noteworthy that IRIS-FGM also applies Seurat objects that can be easily used in the Seurat vignettes.

STROMA4	This package estimates four stromal properties identified in TNBC patients in each patient of a gene expression datasets. These stromal property assignments can be combined to subtype patients. These four stromal properties were identified in Triple negative breast cancer (TNBC) patients and represent the presence of different cells in the stroma: T-cells (T), B-cells (B), stromal infiltrating epithelial cells (E), and desmoplasia (D). Additionally this package can also be used to estimate generative properties for the Lehmann subtypes, an alternative TNBC subtyping scheme (PMID: 21633166).
HPAStainR	This package is built around the HPAStainR function. The purpose of the HPAStainR function is to query the visual staining data in the Human Protein Atlas to return a table of staining ranked cell types. The function also has multiple arguments to personalize to output as well to include cancer data, csv readable names, modify the confidence levels of the results and more. The other functions exist exclusively to easily acquire the data required to run HPAStainR.